Remarks

Examiner continues to reject all claims in prosecution, stating inter alia that Cascio (U.S. 2002/0091818) discloses the detection and resolution of overlapping regions of a character-based user interface (screen) in the process of transforming the interface into a web compatible GUI. Respectfully, Examiner is wrong. Cascio makes no attempt to discern overlapping regions; in fact, the Cascio algorithm precludes any attempt to do so. This will be addressed in great detail below. Examiner is respectfully requested to reconsider these arguments without relying on prior, and erroneous, conclusions.

Applicants vigorously deny that Cascio anticipates the previously amended claims. For example, Cascio does not and cannot detect overlapping interface regions as required in the previous claim1.

The Section 102 Rejection

Cascio's algorithm is explained with reference to Fig. 6 of the patent publication US2002/0091818. Fig. 6 discloses the steps that are applied for identifying a specific character-based interface before performing a transformation of the interface. It is also important to recognize that Fig. 6 applies to identifying a full interface (a full screen) at a time. To this end, a developer has designed a single rule to characterize each screen, each rule consisting of a number of components (text, attributes of text, locations of text on the screen, tables of text, etc) that taken as a whole identifies a screen from all other screens in the legacy application. Each component of a rule is sequentially compared against the screen data until all components are matched, thus identifying this screen out of all other possible screens in the application. If any component of a rule mismatches the current screen data, the screen applying to this rule is eliminated and the next rule for the next possible screen is begun by repeating the steps.

Cascio's Fig. 6 illustrates this algorithm clearly. When a screen's worth of data arrives, step 610 fetches the first rule of a set of rules. Step 620 begins the process of comparing

each component of the rule to the screen data. If the first (or any) component mismatches, step 630 eliminates this potential screen and determines if any more rules (screens) exist for testing. Although Fig. 6 doesn't show it, if all rules have been exhausted, this would correspond to an error condition. If the first component of the first rule matches the screen data, step 650 determines if there are more components in the rule. If so, the next component is tested against the screen data. If and when all components of a rule have been tested and all match the screen data, the screen is identified. Then, step 650 proceeds to step 660 where the screen is identified and the screen data transformed according to the rule.

It is apparent that the Cascio algorithm is incapable of detecting overlapping regions. This is because of the sequential nature of comparing rules and the components of each rule until a screen is identified or eliminated. If all components of a rule match the data, the screen is identified. Otherwise, the screen is eliminated.

Examiner continues to cite a number of paragraphs of Cascio and continues to state that these paragraphs disclose the detection of overlapping regions. Consideration of these paragraphs (par [0058]; par [0072]-[0073]; par [0076]-[0077]) reveals that they have absolutely nothing to do with overlapping regions or detecting overlapping regions. Furthermore, Examiner also asserts that Cascio also resolves the conflicts between overlapping regions. Obviously, this is incorrect. A perusal of the paragraphs cited for this assertion (par [0052]; par [0072]-[0073]; par [0076]-[0077]) also reveals that these paragraphs have nothing to do with overlapped regions.

The Section 103 Rejections

Claims 8-14 are rejected as obvious over Cascio in view of DaCosta. (US Pub 2002/0120725).

Since Cascio does not and cannot disclose, teach or suggest either the detection of overlapping screen regions or the resolution of same, this rejection is improper.

Furthermore, DaCosta discloses a method of updating software applications by attaching to web sites at bootup time and retrieving application updates. The Abstract states that "information can be downloaded to change the graphical user screen for the application". This teaching relates to modifications to an existing GUI, not to a transformation of character screens to a web GUI. Therefore, it is apparent that neither Cascio nor DaCosta contain any disclosure relevant to the present claims of this application.

Summary

Applicants have demonstrated above that the known prior art Cascio and DaCosta do not disclose, teach or suggest the claims in prosecution. Examiner is urged to reconsider the rejection of all claims and pass this application to issue.

Respectfully Submitted,

/Jerry W. Herndon/

Jerry W. Herndon 27,901

IBM Corporation

Intellectual Property Law

Department T81/Building 503

P.O. Box 12195

Research Triangle Park, NC   27709

(919) 542-0345

FAX: 919-542-0345

Email: jherndon@us.ibm.com